

# UN PROGRAMA SIMULADOR DE CIRCUITOS PARA MICROCOMPUTA- DORAS DE USO PERSONAL

*Arturo Prieto Fuenlabrada\*, Mariano Aceves Mijares,  
Tomás Flores\*\**

## Resumen

Este artículo describe un programa que efectúa análisis en frecuencia de una red circuital. El programa está escrito en Basic para una microcomputadora personal. Su manejo es simple y requiere poca memoria; puede ser adaptado a cualquier microcomputadora. La técnica de análisis está basada en el método de la matriz de admitancia indefinida y resuelto por el método de Gauss Jordan modificado. Se presentan asimismo algunos ejemplos.

## Introducción

Existen muchos programas de simulación de circuitos bastante poderosos como el MSINC, SPICE, etc. [1, 2]. Dichos programas pueden ser adaptados a siste-

\* Año sabático otorgado por el Colegio de Electrónica ECFM, Universidad Autónoma de Puebla.

\*\* Instituto Nacional de Astrofísica, Óptica y Electrónica  
Apdo. Postal Nos. 51 y 216  
Puebla, Pue. 72000.

mas computacionales grandes y complejos, sistemas de costo promedio superior a los 3,000 dólares. Sin embargo en nuestro medio es común encontrar sistemas simples de microcomputadoras (TRS-80, Commodore, Sinclair, etc.) de bajo costo, en los que técnicamente resulta bastante difícil adaptar los programas anteriores. Usualmente están escritos en Fortran 77, Fortran IV, etc., mientras la mayoría de las microcomputadoras sólo manejan Basic. El tamaño de la memoria es otro de los impedimentos técnicos para la adaptación [3].

Se intuye así la necesidad de un programa simple escrito en Basic, como el aquí presentado. El objetivo de este trabajo es dotar al profesional o al estudiante de electrónica de una herramienta útil y sencilla, para análisis en frecuencia de redes circuitales, utilizable en cualquier microcomputadora de uso normal que maneje lenguaje Basic.

### Método

El programa usa como técnica de análisis el método de la matriz de admitancia indefinida MAI [4], el cual se muestra en el siguiente ejemplo. Sea una red como la mostrada en la figura 1, la cual consta de tres nodos y tres admitancias, la MAI se forma.

$$\begin{bmatrix} Y_1 + Y_2 & -Y_2 \\ -Y_2 & Y_2 + Y_3 \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

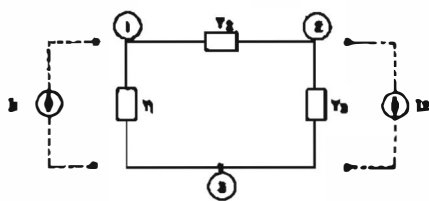


Fig. 1. Red circuital

Donde

[Y] Matriz de admitancia indefinida

V] Vector de voltajes de nodo

I] Vector de fuentes de corriente independiente

El nodo tres no es necesario considerarlo por ser sus relaciones linealmente dependientes de los otros nodos [4]. En caso de existir fuentes de corriente como la mostrada en líneas punteadas, nuestro sistema a resolver será:

$$\begin{bmatrix} Y_1 + Y_2 & -Y_2 \\ -Y_2 & Y_2 + Y_3 \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \end{bmatrix} = \begin{bmatrix} I_1 \\ -I_2 \end{bmatrix}$$

Cabe hacer la aclaración de que las fuentes de voltaje deben ser convertidas a su equivalente Norton para utilizar este método. El sistema se resuelve utilizando el método Gauss Jordan modificado [5]. El diagrama de flujo de programa se muestra en la Fig. 2. Los datos de entrada son almacenados en una serie de arreglos que pueden ser modificados variando su dimensión de acuerdo al tamaño de la memoria de la microcomputadora usada. El programa permite la entrada de los siguientes elementos:

- R. Resistencias valor en ohms.
- C. Capacitores valor en farads.
- L. Inductores valor en henries.
- I. Fuentes de corriente reales en amperes.
- IM. Fuentes de corriente imaginarias en amperes.
- GM. Fuentes de corriente controladas por voltaje en amperes.
- E. Letra para indicar fin de datos de entrada.

Los anteriores elementos deben ser especificados con la letra representativa; además los nodos que son: nodo fuente, nodo destino y valor.

Puede calcular los voltajes de nodo a una frecuencia fija u obtener la función de transferencia de un nodo respecto a otro. En este caso se da la:

- FI, frecuencia inicial
- FF, frecuencia final
- NPF, número de puntos

El programa calcula la magnitud y fase de la función de transferencia. Finalmente se puede obtener una gráfica de los resultados anteriores [6]. El programa se puede considerar interactivo puesto que va guiando cada paso del usuario.

### Ejemplos

#### Ejemplo 1.

Se desean conocer los voltajes de nodo de la red mostrada en la figura 3, donde  $I = 2$  amper,  $R_1 = 5$  ohms,  $R_2 = .5$  ohms,  $C = 1$  farad.

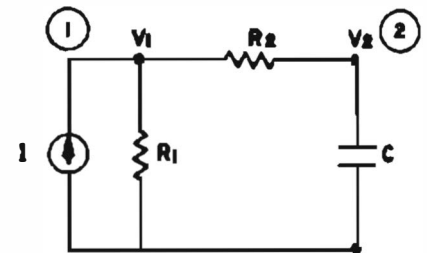


Fig. 3. Circuito del ejemplo 1

Note que la fuente I con  $R_1$  y  $R_2$  es equivalente a una fuente de voltaje de 1 voltio con una resistencia en serie de 1 ohm.

Después de iniciar con "Run" el programa, escribirá el título y preguntará por:

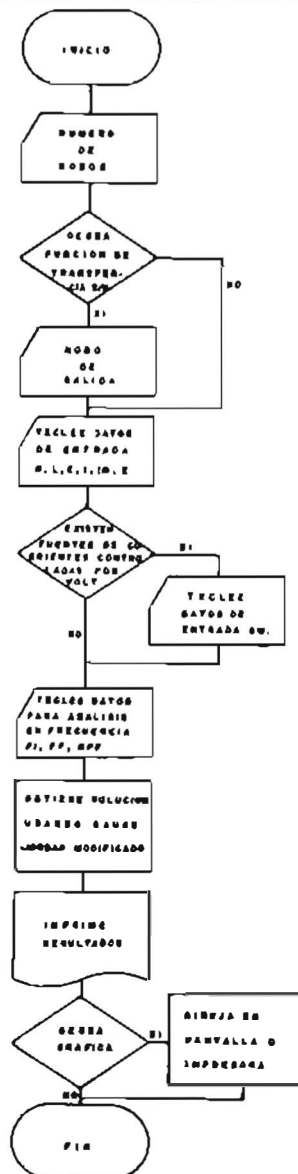


Fig. 2. Diagrama de flujo

```

SOLUCION DE REDES
TECLEE LOS ELEMENTOS
TIPO (R,C,L,I,IM,E),MODOS VALOR
I 0 1 2
R 0 1 .5
R 1 2 .3
C 0 2 1
E 0 0 0
FUENTES CONTROLADAS Y MODOS DE CONTROL
E 0 0 0 0
DA LA FREC.INICIAL FINAL Y NUMERO DE PUNTOS
0 .25 20
FREC(KZ)MAG(DB)
0 -5.60711956E-09
.0125 -.0247071772
.025 -.105837193
.0375 -.234650331
.05 -.40877534
.0625 -.622857892
.075 -.87094683
.0875 -1.14494561
.1 -1.44504988
.1125 -1.73989514
.125 -2.08649771
.1375 -2.42140904
.15 -2.74062739
.1625 -3.10157116
.175 -3.44200874
.1875 -3.78018459
.2 -4.11474294
.2125 -4.44465891
.225 -4.74917717
.2375 -5.08775931

```

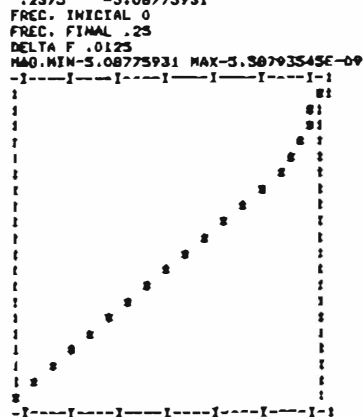


Figura 4

(es necesario completar con ceros para finalizar).

¿Ahora las fuentes controladas y sus nodos de control?

E, 0, 0, 0, 0

(en este ejemplo no hay fuentes controladas).

Da la frecuencia inicial, final y número de puntos.

0, .5, 2 0

El programa calcula los voltajes de nodo y al final pregunta:

¿Desea gráfica s/n? S

¿Ancho de pantalla? 32

(nuestra micro sólo permite 32 caracteres horizontales).

Los resultados se muestran en la figura 4.

Ejemplo 2.

Encontrar la función de transferencia del circuito mostrado en la figura 5, donde  $V_{in} = 0.001666$  Amp,  $R_{in} = 600$  ohms,  $r_{\pi} = 1$  Kohm,  $C_{\pi} = 5$  pf,  $C_{\mu} = 15$  pf,  $G_M = 1$  ohm,  $R_E = 100$  ohms,  $R_L = 1$  Kohm.

También los resultados se muestran en la figura 6. El procedimiento es idéntico al ejemplo 1, con la diferencia de que incluye una fuente de corriente controlada por el voltaje  $V_{\pi}$ .

```

SOLUCION DE REDES
TECLEE LOS ELEMENTOS
TIPO (R,C,L,I,IM,E),MODOS VALOR
I 0 1 1.6444E-03
R 0 1 400
R 1 2 1000
C 1 2 5E-12
C 1 3 1.5E-11
R 2 0 100
R 3 0 1000
E 0 0 0
FUENTES CONTROLADAS Y MODOS DE CONTROL
GM 3 2 .1 1 2
E 0 0 0 0
DA LA FREC.INICIAL FINAL Y NUMERO DE PUNTOS
1000 100000 20
FREC(KZ)MAG(DB)
1000 18.4339301
3950 18.4338703
10900 18.4337233
15850 18.4334932
20800 18.4331797
25730 18.4327792
30700 18.4322934
35630 18.4317220
40600 18.4310671
45530 18.4303263
50500 18.4315004
55430 18.4303896
60400 18.4293394
65350 18.4283125
70300 18.4273482
75250 18.4264083
80200 18.4247638
85150 18.4233448
90100 18.4218411
95050 18.4202534

```

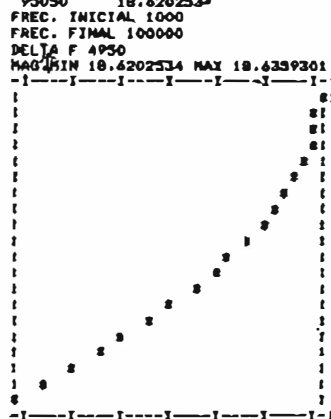


Figura 6

Conclusiones

En resumen, el programa desarrollado (figura 7) simula en forma correcta redes circuitales. El empleo del programa es sencillo y transparente para el usuario. Este programa fue escrito en una TRS 80 Microcolor Computer modelo MC10 de 16K permitiendo hasta 30 nodos de análisis. El programa es bastante versátil y su adaptación a una Sinclair 2068 fue prácticamente directo.

Figura 7

```

5 PRINT 'SOLUCION DE REDES '
6 PRINT 'SOLUCION DE REDES'
10 INPUT 'NUMERO DE MODOS' IN
12 DIM G(N+1),B(N+1),TIP(110),NF(110),HD(110),UL(110),MA(110),MB(110)
13 DIM F(110),VM(110),UF(110)
15 INPUT 'DEBEAS FUNCION DE TRANSFERENCIA?S/N' IAB
20 IF AB='B' THEN GOTO 30
25 A=1
26 GOTO 32
30 INPUT 'EL MODO DE SALIDA' IU
32 PRINT 'TECLEE LOS ELEMENTOS'
33 LPRINT 'TECLEE LOS ELEMENTOS'
35 PRINT 'TIPO(R,C,L,I,IM,E),MODOS VALOR'
36 LPRINT 'TIPO (R,C,L,I,IM,E),MODOS VALOR'
40 L=1
45 INPUT TIP(L),NF(L),HD(L),UL(L)

```

Continúa

¿Números de nodos? 2  
 ¿Desea función de transferencia s/n? S  
 ¿Nodo de salida? 2

(si no pide función de transferencia imprimirá todos los voltajes de nodo).

Teclée los elementos

Tipo (R, C, L, I, IM, E), nodos, valor.

I, 0, 1, 2

.

.

E, 0, 0, 0

Continuación de la fig. 7

```

46 LPRINT TIP*(L)JNF(L)JND(L)JUL(L)
40 IF TIP*(L)="E" THEN GOTO 49
42 L=L+1
45 GOTO 45
49 L=L+1
70 PRINT 'AHORA LAS FUENTES CONTROLADAS Y BUS
MODOS DE CONTROL'
71 LPRINT 'FUENTES CONTROLADAS Y MODOS DE CONTROL'
75 INPUT TIP*(L),NF(L),ND(L),UL(L),MA(L),NB(L)
80 LPRINT TIP*(L)JNF(L)JND(L)JUL(L)JMA(L)JNB(L)
85 IF TIP*(L)="E" THEN GOTO 95
87 L=L+1
90 GOTO 75
95 PRINT 'DA LA FREQ.INICIAL FINAL Y NUMERO DE PUNTOS'
96 LPRINT 'DA LA FREQ.INICIAL FINAL Y NUMERO DE PUNTOS'
100 INPUT FI,FF,NPF
101 LPRINT FI,FF,NPF
105 DELF=(FF-FI)/NPF
110 IF A=1 THEN FF=FI
115 FOR F=FI TO FF STEP DELF
120 DM=2.3145927*F
125 FOR B=1 TO L-1
130 IF TIP*(B)="R" THEN GOSUB 2000
135 IF TIP*(B)="C" THEN GOSUB 3000
140 IF TIP*(B)="L" THEN GOSUB 2200
145 IF TIP*(B)="I" THEN GOSUB 2300
150 IF TIP*(B)="M" THEN GOSUB 2400
155 IF TIP*(B)="G" THEN GOSUB 2500
165 NEXT B
220 FOR I=1 TO N
225 AR=D(I,I)
230 A1=B(I,1)
235 A2=(AR*244I^2)
240 IF A1=0 THEN 360
245 FOR J=1 TO N+1
250 D(I,J)=D(I,J)+AR+R(I,J)*A1)/AT
255 B(I,J)=B(I,J)+AR/AT-D(I,J)*A1/AR+B(I,J)*A1^2)/(AT*AR)
260 NEXT J
265 FOR K=1 TO M
270 IF (K-1)=0 THEN 305
275 BR=D(K,I)
280 BI=B(K,1)
285 FOR J=1 TO N+1
290 G(K,J)=G(K,J)-(BR*G(I,J)-BI*B(I,J))
295 B(K,J)=B(K,J)-(BR*B(I,J)+BI*G(I,J))
300 NEXT J
305 NEXT K
310 NEXT I
315 IF U=0 THEN GOTO 326
319 M=U
320 PRINT 'F=';F
321 PRINT 'U=';M;I;J;K;L;M;N;O;P;Q;R;S;T;U;V;W;X;Y;Z
323 GOTO 336
325 LPRINT 'LOS VOLTAJES DE NODO 80M'
326 PRINT 'LOS VOLTAJES DE NODO 50M'
327 FOR M=1 TO N
330 PRINT 'U(';M;N;1;')=';D(M,N+1);'+';B(M,N+1);'+';J'
331 LPRINT 'U(';M;N;1;')=';D(M,N+1);'+';B(M,N+1);'+';J'
332 NEXT M
335 GOTO 354
336 X=D(M,N+1);Y=B(M,N+1)
337 Z=Z+1
338 UV=80R*(X*Y+Y*Y)
339 UM(Z)=20.8.43429448*LOG(UV)
347 F(Z)=F
348 FOR U=1 TO M
349 FOR UV=1 TO N+1
350 G(U,UV)=D(U,UV)+B(U,UV)=0
351 NEXT UV
352 NEXT U
354 NEXT F
355 GOTO 365
360 PRINT 'LA DIAGONAL PRINCIPAL ES CERO'
361 LPRINT 'LA DIAGONAL PRINCIPAL ES CERO'
365 PRINT 'FREQ.(M2)=';MAQ(DB)
366 LPRINT 'FREQ.(M2)=';MAQ(DB)
367 FOR Z=1 TO NPF
370 PRINT F(Z);TAB(10);UM(Z)
371 LPRINT F(Z);TAB(10);UM(Z)
375 NEXT Z
380 INPUT 'DESEA GRAFICA S/N';AA
381 IF AA="N" GOTO 410
385 INPUT 'ANCHO DE PANTALLA';IL
390 PRINT 'FRECUENCIA INICIAL';FI
391 LPRINT 'FREQ. INICIAL';FI
395 PRINT 'FRECUENCIA FINAL';FF
396 LPRINT 'FREQ. FINAL';FF
400 PRINT 'DELTA F';DELFF
401 LPRINT 'DELTA F';DELFF
405 GOSUB 4000
410 END
1000 FOR M=1 TO N
1010 FOR I=1 TO M+1
1020 PRINT U(M,I);B(M,I);'+';J '
1030 NEXT I
1035 PRINT
1036 NEXT M
1040 PRINT 'ESTAN CORRECTOS SUS DATOS S/N'
1042 INPUT L$
1043 RETURN
2000 NF=NF(B)
2001 ND=ND(B)
2002 UL=UL(B)
2005 IF (NF<>0) AND (ND<>0) THEN 2060
2010 IF NF=0 THEN 2030
2020 ND=NF
2021 G(ND,ND)=G(ND,ND)+1/UL
2023 RETURN
2030 NF=ND

```

```

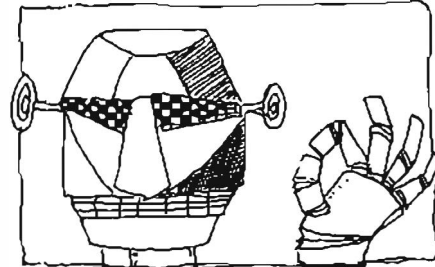
2040 G(NF,NF)=G(NF,NF)+1/UL
2050 RETURN
2060 G(ND,ND)=G(ND,ND)+1/UL
2070 G(NF,NF)=G(NF,NF)+1/UL
2080 G(NF,ND)=G(NF,ND)+1/UL
2090 G(ND,NF)=G(ND,NF)-1/UL
2095 RETURN
2200 NF=NF(B)
2201 ND=ND(B)
2202 UL=UL(B)
2205 IF (NF<>0) AND (ND<>0) THEN 2260
2210 IF NF=0 THEN 2230
2220 ND=NF
2221 B(ND,ND)=B(ND,ND)-1/(UL*DM)
2223 RETURN
2230 NF=ND
2240 B(NF,NF)=B(NF,NF)-1/(UL*DM)
2250 RETURN
2260 B(ND,ND)=B(ND,ND)-1/(UL*DM)
2270 B(NF,NF)=B(NF,NF)-1/(UL*DM)
2280 B(NF,ND)=B(NF,ND)+1/(UL*DM)
2290 B(ND,NF)=B(ND,NF)+1/(UL*DM)
2295 RETURN
2300 NF=NF(B)
2301 ND=ND(B)
2302 UL=UL(B)
2305 IF (NF<>0) AND (ND<>0) THEN 2360
2310 IF NF=0 THEN 2330
2320 ND=NF
2321 G(ND,N+1)=G(ND,N+1)-UL
2323 RETURN
2330 NF=ND
2340 G(NF,N+1)=B(NF,N+1)+UL
2350 RETURN
2360 G(ND,N+1)=B(ND,N+1)+UL
2365 G(NF,N+1)=B(NF,N+1)-UL
2375 RETURN
2400 NF=NF(B)
2401 ND=ND(B)
2402 UL=UL(B)
2405 IF (NF<>0) AND (ND<>0) THEN 2460
2410 IF NF=0 THEN 2430
2420 ND=NF
2421 B(ND,N+1)=B(ND,N+1)-UL
2423 RETURN
2430 NF=ND
2440 B(NF,N+1)=B(NF,N+1)+UL
2450 RETURN
2460 B(ND,N+1)=B(ND,N+1)+UL
2465 B(NF,N+1)=B(NF,N+1)-UL
2475 RETURN
2500 ND=ND(B)
2501 NF=NF(B)
2502 MA=MA(B)
2503 NB=NB(B)
2504 UL=UL(B)
2505 IF (NF<>0) AND (NB<>0) THEN 2560
2510 IF NF=0 THEN 2530
2520 ND=NF
2521 G(ND,MA)=G(ND,MA)+UL
2522 G(ND,NB)=G(ND,NB)-UL
2523 RETURN
2530 NF=ND
2540 G(NF,MA)=G(NF,MA)-UL
2545 G(NF,NB)=G(NF,NB)+UL
2550 RETURN
2560 G(NF,MA)=B(NF,MA)+UL
2570 G(NF,NB)=B(NF,NB)-UL
2580 G(ND,MA)=G(ND,MA)+UL
2590 G(ND,NB)=G(ND,NB)+UL
2595 RETURN
3000 NF=NF(B)
3001 ND=ND(B)
3002 UL=UL(B)
3005 IF (NF<>0) AND (NB<>0) THEN 3060
3010 IF NF=0 THEN 3030
3020 ND=NF
3021 B(ND,ND)=B(ND,ND)+UL*DM
3023 RETURN
3030 NF=ND
3040 B(NF,NF)=B(NF,NF)+UL*DM
3050 RETURN
3060 B(ND,ND)=B(ND,ND)+UL*DM
3070 B(NF,NF)=B(NF,NF)+UL*DM
3080 B(NF,ND)=B(NF,ND)-UL*DM
3090 B(ND,NF)=B(ND,NF)-UL*DM
3095 RETURN
4000 B=100000000
4005 FOR I=1 TO NPF
4010 IF B>UM(I) THEN B=UM(I)
4015 NEXT I
4020 FOR I=1 TO NPF
4025 UM(I)=UM(I)-B
4030 NEXT I
4035 C=0
4040 FOR I=1 TO NPF
4045 IF C<UM(I) THEN C=UM(I)
4050 NEXT I
4055 A=L/C
4060 E=A*ABS(B)
4065 PRINT 'MAG. MIN';B; 'MAX';C;B
4066 LPRINT 'MAG.MIN';B; 'MAX';C;B
4070 IF B>0 THEN GOTO 4085
4075 IF ABS(B)>C THEN GOTO 4085
4080 PRINT TAB(E);'0'
4081 LPRINT TAB(E);'0'
4085 GOSUB 5000
4090 FOR I=1 TO NPF
4115 E2=ABSUM(I)
4120 IF E2="1" THEN GOTO 4135
4125 PRINT '0'
4126 LPRINT '0'

```

```

4130 GOTO 4150
4135 PRINT '0'
4136 LPRINT '0'
4140 PRINT TAB(E2);'0'
4141 LPRINT TAB(E2);'0'
4150 PRINT TAB(L);'1'
4151 LPRINT TAB(L);'1'
4155 NEXT I
4160 GOSUB 5000
4165 RETURN
5000 E3=E-5*INT(E/5)
5005 E3=E-5*INT(E/5)
5005 IF B>0 THEN E3=0
5010 IF ABS(B)>C THEN E3=0
5015 FOR J=1 TO E3
5020 PRINT '0'
5021 LPRINT '0'
5025 NEXT J
5030 FOR J=1 TO (L-E3)/5
5040 PRINT '1-----1'
5041 LPRINT '1-----1'
5045 NEXT J
5050 PRINT '1'
5051 LPRINT '1'
5055 E4=(J-1)*5+1+E3
5065 IF E4=L+1 THEN GOTO 5095
5070 E4=E4+1
5075 IF E4=L+1 THEN GOTO 5090
5080 PRINT '0'
5081 LPRINT '0'
5085 GOTO 5070
5090 PRINT '1'
5091 LPRINT '1'
5095 RETURN

```



Se debe buscar lo óptimo en lo que respecta a cantidad de memoria, según la microcomputadora con que se cuente (i.e., variar dimensiones de los arreglos de entrada). El sistema es relativamente lento comparado con un sistema grande, pero por tratarse de un solo usuario sin tiempo compartido, son comparables en lo que a eficiencia se refiere.

Debido a la estructura y método de análisis es sencillo incluir algunos modelos de biblioteca como el modelo híbrido del transistor, amplificadores operacionales, etcétera.

Referencias

1. SPICE. 2. A computer program to simulate semiconductor circuits, ERL MEMO No. ERL-M520 University of California, Berkeley, 1985.
2. MSINC A program for simulation of MOS integrated circuits, el manual se puede conseguir en el Centro de Computo del INAOE.
3. Integrating microprocessors into product design. A.K. Kochhar, N.D. Burns, Marcel Dekker, Inc. New York and Basel, 1983.
4. Introduction to circuit analysis, Trick-Ti-monthly, Wiley, 1977.
5. Computer programs for the electronic analysis and design, Dimitri S. Bug-Nolo, Reston Publishing Co., 1983.
6. Basic scientific subroutines, F.R. Ruckdeschel, Mc-Graw Hill, 1981.